APPLICATION FOR

UNITED STATES LETTERS PATENT

SPECIFICATION

INVENTOR(s): Hiroaki KIMURA, Yuji SHIRAHIGE
and Iwao YAMAZAKI

Title of the Invention: CACHE CONTROL DEVICE AND METHOD

CACHE CONTROL DEVICE AND METHOD

Background of the Invention

Field of the Invention

The present invention relates to a computer system provided with key control protection by a storage key. More particularly, it relates to a cache control device for controlling the operations of the cache, and a method thereof.

Description of the Related Art

A storage key is information for protecting the contents of a main storage of an information processing device from an improper access, and the key is created for each page of the main storage. This storage key consists of, for example, an access control bit (four bits), a fetch protection bit, a reference bit, and a change bit.

Figure 1A is a block diagram showing the constitution of a conventional information processing unit that has a storage key. The information processing unit of Figure 1A includes a CPU (Central Processing Unit) instruction computation unit 31, a CPU memory unit 32, a main storage 33, and a key storage 41. The CPU memory unit 32 includes two caches 34, a control device

35, a main memory access validity detection circuit 36, two TLBs (Translation Lookaside Buffers) 37, two key buffers 38, a DAT (Dynamic Address Translation) 39, and a key access port 40.

TLB 37 is provided to speed up address translation, and the correspondence relation between a logical address and a physical address is registered in TLB 37. The key buffer 38 is provided to speed up a key access, and a storage key is registered in the key buffer 38. Here, the key accessing process demands a storage key stored in the key storage 41, thereby obtaining the key. Further, the cache 34 is provided to speed up a main memory access, thereby storing data.

One of the two caches 34 stores instructions as data, while the other stores operands as data. Regarding TLBs 37 and key buffers 38, one is used for instructions, while the other is used for operands. DAT 39 transforms an assigned logical address into a physical address.

The validity detection circuit 36 checks the validity of the main memory access, using a storage key registered in the key buffer 38 and an access key transmitted from the CPU instruction computation unit 31. The Key access port 40 stores a storage key transmitted from the key storage 41, and transfers it to the key buffer 38.

Figure 1B is a timing chart of the cache control in the information processing unit of Figure 1A. Figure 1C is a flowchart showing the outline of the cache control. When a main memory access demand (request) is issued from the CPU instruction computation unit 31, it is determined whether the physical address that is a translation result of the logical address associated with the demand is registered in TLB 37 (hit or miss) (step S11).

In the case that the physical address is not registered, TLB 37 reports a TLB miss to the control device 35, and then the control device 35 demands address translation from DAT 39(step S12). After the address translation process is carried out by DAT 39, the control device 35 demands a storage key from the key storage 41(step S13), and waits for a key arrival report (step S13).

When the storage key arrives at the key access port 40, this arrival triggers the registration of a physical address that is the translation result of DAT 39, in TLB 37, and also the registration of a storage key of the key access port 40, in the key buffer 38 (step S15). Then, DAT 39 is released (step S16).

Here, the control device 35 resumes a process of a main memory access demand. Since the physical address

is registered in TLB 37, TLB 37 reports a TLB hit to the control device 35, and the control device 35 retrieves the cache 34 using this physical address which is read out from TLB 37. In this way, it is determined whether data (instruction or operand) is registered (hit or miss) in the cache 34 (step S17).

If data is not registered, a data demand is issued to the main storage 33, and the arrived data is registered in the cache 34 (step S18). Then, a process of the main memory access demand is resumed. If data is registered, the cache 34 transmits the data, and reports a cache hit to the control device 35.

The CPU instruction computation unit 31 determines in advance whether a main memory access demand requires protection (key control protection) using a storage key, and it transmits the demand that differs according to the determination result to the CPU memory unit 32. Then, the CPU memory unit 32 determines whether a key check is required by the validity detection circuit 36 on the basis of a type of the demand (step S19). At the same time as the TLB hit, the validity detection circuit 36 reads out a storage key that is registered in the key buffer 38, and it checks the validity of access by comparing this key with an access key received from the CPU instruction computation unit 31 (step S20).

Specifically, in the case that the key control protection is not required, the CPU instruction computation unit 31 transmits to the CPU memory unit 32, a signal INH_KEY_CHECK for disregarding the check result of the validity detection circuit 36. The CPU memory unit 32 is configured to disregard the check result when this signal is on (logic value 1).

If the main memory access demand hits a cache, and the validity of access is admitted, data is transmitted to the CPU instruction computation unit 31 (step S21). The control device 35 transmits a data transfer report and a completion report to the CPU instruction computation unit 31, and completes the main memory access process. At this time, the check result of the validity detection circuit 36 is transmitted to the CPU instruction computation unit 31 as a key protection exception signal. If the key protection exception signal is off (logic value 0) when the completion report is transmitted, the transmitted data comes into effect.

If the validity of access is not admitted at step S20, the key protection exception signal becomes on, an exception report is transmitted to the CPU instruction computation unit 31 (step S22).

However, the above-mentioned conventional cache control has the following problems.

In conventional control, a main memory access is executed to the main memory access demand that requires key control protection, the validity of access is checked for the obtained data, and the checked data is transmitted after the validity of access is admitted. Because of this, in the case that a key access is required for the main storage, the device should wait for the arrival of the storage key, so that it takes a long time to transmit data.

Further, the arrival of the storage key that is required for the main storage at a key access port triggers the registration of the translation results obtained by DAT, in the TLB, and also the release of DAT. Because of this, neither the registration of the translation results nor the release of the DAT can be executed, even if DAT completes an address translation process.

In the main memory access demand that does not require key control protection, the validity check of access is not basically required. In spite of this, when a TLB miss occurs in such a demand, a key access is executed to the main storage, and data is checked by using the arrived storage key, thereby transmitting data. Therefore, a useless waiting time occurs.

Summary of the Invention

The object of the present invention is to provide a cache control device for improving the performance of a data access in an information processing unit that is provided with key control protection by a storage key, and a method thereof.

The cache control device of the present invention is provided with a demand device, a control device, and a transfer device and a cache stores data.

The demand device demands a storage key, and the control device starts the retrieval of data of the cache without waiting for the arrival of the demanded storage key. Then, the transfer device transmits the retrieved data.

Brief Description of the Drawings

Figure 1A is a block diagram showing the constitution of a conventional information processing unit;

Figure 1B is a timing chart of conventional cache control;

Figure 1C is a flowchart of conventional cache control;

Figure 2A is a block diagram showing the principle of a cache control device of the present invention;

Figure 2B is a block diagram showing the constitution of an information processing unit;

Figure 3 is the first timing chart of the cache control;

Figure 4 is the second timing chart of the cache control;

Figure 5 is a flowchart of cache control;

Figure 6 is a block diagram showing the circuit of a CPU memory unit;

Figure 7 is a block diagram showing the first input-output signal of DAT;

Figure 8 is a block diagram showing the first input-output signal of a key access port;

Figure 9 is a block diagram showing the constitution of a control device;

Figure 10 is a diagrammatic illustration of the second input-output signal of DAT;

Figure 11 is a diagrammatic illustration of an input-output signal of the control device;

Figure 12 is a block diagram showing the second input-output signal of the key access port;

Figure 13 is a diagrammatic illustration of the stage of an instruction process;

Figure 14 illustrates single control;

Figure 15 illustrates pipeline control;

Figure 16 is a diagrammatic illustration of DAT release; and

Figure 17 is a diagrammatic illustration of TLB registration.

Description of the Preferred Embodiments

The preferred embodiments of the present invention are explained in detail in reference to the drawings.

Figure 2A is a block diagram of the cache control device of the present invention. The cache control device of Figure 2A is provided with a cache 51, a demand device 52, a control device 53, and a transfer device 54.

The cache device 51 stores data, the demand device 52 demands a storage key, and the control device 53 starts the retrieval of data of the cache device 51, without waiting for the arrival of the demanded storage key. Then, the transfer device 54 transmits the retrieved data.

In the case that TLB retrieval is carried out for a data access demand and a storage key is required by the TLB miss, the demand device 52 issues a key demand for a storage unit of the storage key in a similar manner to the key storage 41, and demands the corresponding storage key. When the storage key is demanded, the control device 53 starts the retrieval of the cache 51, by

retrieving TLB again before the arrival of the demanded storage key. Then, the transfer device 54 immediately transmits the retrieved data to a demand source of the data access demand.

According to such constitution, the process of a main memory access demand can be resumed, without waiting for the completion of a validity check executed using the arrived storage key, in the case that a key access is required for the main storage. Thus, the process time of a demand for the main memory access can be shortened. Further, the transfer device 54 can transmit the retrieved data, without waiting for the arrival of the demanded storage key. Accordingly, the time required for the issuance of a main memory access demand for the data transfer can be shortened.

The cache 51 of Figure 2A corresponds to, for example, the cache 34 of Figure 2B, which is described later. The demand device 52 and the control device 53 of Figure 2A correspond to, for example, a control device 61 of Figure 2B. The transfer device 54 of Figure 2A corresponds to, for example, an AND gate 84 of Figure 6, which is also described later.

The following are the characteristics of the cache control device of the present embodiment.

(1) The validity check of a main memory access and a

cache access are separated, and data are processed before the validity check. Specifically, in the case that a key access is required for the main storage, a CPU memory unit is controlled to transmit data, without waiting for the arrival of a storage key at a key access port and the validity check. In this way, the time required for the issuance of a main memory access demand for the transfer of data can be shortened.

(2) In the case that a TLB miss of a main memory access occurs, address translation at DAT is required, and the requirement of a storage key for the main storage is treated as a trigger for the registration of the result of address translation by DAT in TLB. In this way, the time required from the start of address translation to the TLB registration can be shortened, thereby retrieving TLB again without waiting for the arrival of the storage key. That is, the process of a main memory access demand can be resumed without waiting for a key access.

(3) After the result of address translation by DAT is registered in TLB, DAT is released without waiting for the registration of a storage key, and the successive address translation demands are processed. In this case, the demand of a storage key for the main storage is treated as a trigger to release DAT. In this way, DAT is immediately released, and the successive address translation

processes are performed without waiting for the arrival of the storage key after the address translation is executed.

(4) A main memory access process is completed without waiting for a key access and a validity check executed for the main memory access demand that does not require key control protection. In this case, the data and completion report are immediately transmitted, if the demanded data exists in the cache even if a TLB miss occurs. Then, the storage key arrived from the main storage is registered in a key buffer. Therefore, the time between the issuance of a main memory access demand and that of a completion report can be shortened.

(5) Since the successive demands should be waited until the key access port becomes vacant only by performing the above-mentioned control (3), key accesses cannot continuously be executed in the case that a plurality of storage key demands continuously occur. Thereupon, a plurality of key access ports are prepared and a plurality of key accesses are simultaneously executed, so that the release of DAT can efficiently be executed. In this way, since the address translation by DAT is continuously carried out and a corresponding storage key can be demanded each time the translation completes, the key demand for the main storage is continuously issued.

Accordingly, DAT works efficiently.

Figure 2B is a block diagram showing the configuration of an information processing unit equipped with a cache control device of the present invention. In Figure 2B, the same item number shown in Figure 1A indicates the identical or corresponding function, regarding the constitution element of the same number as shown Figure 1A. As mentioned above, the control device 61 performs control to make a data access of the CPU instruction computation unit 31 efficient, and two key access ports 62 are provided.

Each of Figures 3 and 4 shows a timing chart of the cache control in the information processing unit of Figure 2B. Figure 5 shows a flowchart indicating the outline of the cache control. A timing chart of Figure 3 corresponds to the case that key control protection is required, while a timing chart of Figure 4 corresponds to the case that key control protection is not required.

First of all, a main memory access demand is issued from the CPU instruction computation unit 31, and a logical address associated with the demand is delivered to a control device 61. The control device 61 accesses TLB 37 in order to check whether the physical address that is a translation result of the logical address is registered in TLB 37 (step S31). If the physical address

is registered in TLB 37, TLB 37 reports a TLB hit to the control device 61.

If the physical address is not registered, the TLB 37 reports a TLB miss to the control device 61, and the control device 61 demands address translation from DAT 39 (step S32). Following this, DAT 39 starts address translation of the logical address. When the address translation process completes, the logical address and physical address are stored in a register provided inside the control device 61.

Further, the control device 61 issues a storage key demand to the key storage 41 (step S36), and waits for a key arrival report (step S33). Furthermore, this storage key demand is treated as a trigger for the issuance of a demand for registration of a physical address that is a translation result of DAT 39 (TLB registration demand) in TLB 37 (step S34), and for the registration of the translation result in TLB 37, thereby releasing DAT 39 (step S35).

Next, making the TLB registration demand as a trigger, the control device 61 resumes a process of the main memory access demand, and accesses TLB 37 again using the logical address stored in the CPU memory unit 32. Since the translation result is registered in TLB 37 this time, TLB 37 reports a TLB hit to the control

device 61. The control device 61 that receives a report of a TLB hit retrieves the cache 34 using the thus-hit physical address (step S38).

If the demanded data is not registered in the cache 34A, a data demand is issued to the main storage 33, and the arrived data is registered in the cache 34 (step S39). If data is registered, it is checked whether the key check by the validity detection circuit 36 is required (whether key control protection is required) (step 40), and it is also checked whether data has been transmitted or not (step S45).

If the key check is required, it is determined whether a validity signal VLD of the key buffer 38 is on (step S41). If the signal VLD is off, the storage key has not arrived, and accordingly the arrival is waited for (step S36).

If data has not been transmitted, the cache 34 transmits data to the CPU instruction computation unit 31, and it reports a cache hit to the control device 61 (step S46). The control device 61 that receives the report of a cache hit transmits a data transfer report to the CPU instruction computation unit 31.

Then, a storage key is delivered to a key access port 62 from the main storage 33. Thereupon, making the arrival as a trigger, the control device 61 issues a

demand of registering a storage key in the key buffer 38 (KEY registration demand), and a storage key of the key access port 62 is registered in the key buffer 38 (step S37). In this way, VLD of the key buffer 38 becomes on.

At this time, the logical address and physical address that are registered in TLB 37, and the storage key should be corresponded. Thereupon, the control device 61 overwrites on TLB 37, the logical address and physical address that are stored in a register inside the control device 61 at the same time as the registration of the storage key.

As shown in Figure 3, the registration of the storage key triggers the start of a process of the main memory access demand in the CPU memory unit 32. In a manner similar to the above-mentioned procedure, a TLB hit and a cache hit occur. In this case, since VLD is on, the key buffer 38 is accessed, the registered storage key is read out, and the validity of access is checked by the validity detection circuit 36 (step S42).

If the validity of access is not admitted as a result of the key check executed by the validity detection circuit 36, the control device 61 turns a key protection exception signal on, transmits an exception report to the CPU instruction computation unit 31 (step S43), transmits a completion report (step S44), and completes

a main memory access process. If the validity of access is admitted, the control device 61 turns the key protection exception signal off, and performs a process at step S44.

In the case that a key check executed by the validity detection circuit 36 is not required, a completion report is transmitted to the CPU instruction computation unit 31 at the same time as the data transfer report, as shown in Figure 4 (steps S44 and S46). In this case, the result of the key check is not referred to, and the process of the main memory access demand is not resumed after the registration of the storage key.

Next, with reference to Figures 6 to 12, the circuit constitution of the CPU memory unit 32 of Figure 2B is explained in detail in the case that the cache control as shown in Figure 3 is carried out.

Figure 6 shows the connection relation of circuits of the CPU memory unit 32. Inverters 77 and 79, AND gates 78, 80, and 81, and an OR gate 82 of Figure 6 belong to the control device 61.

A signal LA_TO_TLB expresses a logical address transmitted from the control device 61 to TLB 37. A signal LA_FROM_DAT expresses a logical address output from DAT 39. A signal AA_FROM_DAT expresses a physical address output from DAT 39.

An OR gate 71 outputs to TLB 37, the OR operation of a TLB registration demand signal and a key registration demand signal, which are output from the control device 61. A comparator 72 compares the signal LA_TO_TLB and a logical address LA that is output from TLB 37, and it outputs a signal that turns on when the two agree. An inverter 73 reverses the output of the comparator 72, and generates a signal TLB_MISS that turns on in the case of a TLB miss. An AND gate 74 outputs the AND operation of the output of the comparator 72 and a physical addresses AA that is output from TLB 37.

As mentioned above, a signal INH_KEY_CHECK is the one that is sent from the CPU instruction computation unit 31 to the CPU memory unit 32, and the signal turns on in the case that key control protection is not required. A signal ACCESS_KEY expresses an access key that is sent from the CPU instruction computation unit 31 to the CPU memory unit 32. A signal LA_TO_KEY_BUFFER expresses a logical address that is sent from the control device 61 to the key buffer 38. A signal KEY_FROM_KEY_PORT shows a storage key that is output from the key access port 62.

The validity detection circuit 36 includes a comparator 75 and an OR gate 76. The comparator 75 compares the signal ACCESS_KEY and the storage key that is output

from the key buffer 38, and it outputs a signal that turns on when the two agree. The OR gate 76 outputs the OR operation of the signal INH_KEY_CHECK and the output of the comparator 75. An inverter 77 reverses the output of the OR gate 76, and generates a key protection exception signal that turns off in the case that the transmitted data is effective.

A signal LA_TO_CACHE expresses the logical address that is sent from the control device 61 to the cache 34. A comparator 83 compares the output of the AND gate 74 and the physical address AA that is output from the cache 34, and it outputs a signal that turns on when the two agree. The AND gate 78 outputs the AND operation of a validity signal VLD that is output from the key buffer 38 and the output of the comparator 83, as a completion report.

The inverter 79 reverses the signal VLD to be output. The AND gate 80 outputs the signal VLD, the AND operation of the output of the comparator 83 and a data unsent signal. The AND gate 81 outputs the output of the inverter 79, and the AND operation of the output of the comparator 83 and a data unsent signal. The data unsent signal is on before data is transmitted, while the signal turns off once data is transmitted. The OR gate 82 outputs the OR operation of the outputs of the AND gates 80 and

81, as a data transfer report.

Further, the AND gate 84 outputs the output of the comparator 83, and the AND operation of the data (DATA) and the validity signal VLD which are output from the cache 34, as data.

According to such circuit constitution, data is immediately transmitted to the CPU instruction computation unit 31, if the physical address that is read out by the TLB hit agrees with the physical address of the corresponding cache 34. At the same time, a data transfer report is transmitted without considering whether a storage key is registered in the key buffer 38. Further, a completion report is immediately transmitted when a storage key is registered in the key buffer 38 and the signal VLD becomes on.

Figure 7 shows an input-output signal of DAT 39. An address translation demand signal is output from the control device 61. A signal LA_TO_DAT shows the logical address that is sent from the control device 61 to DAT 39. DAT 39 transforms the signal LA_TO_DAT into a physical address AA when an address translation demand signal turns on to be output, and it turns an address translation completion signal on.

A physical address AA of the translation result is stored in a register 85 provided in the control device

61 to be output as a signal AA_FROM_DAT. At this time, even the signal LA_TO_DAT is stored in the register 85 to be output as a signal LA_FROM_DAT. A signal DAT_BUSY turns on during use of DAT 39, and turns off when the translation result is registered in TLB 37.

Figure 8 shows the input-output signal of the key access port 62. A signal KEY_PORT_VALID is output from the control device 61, and a signal KEY_FROM_KEY_STORAGE shows the storage key that is transmitted from the key storage 41. The key access port 62 outputs the key KEY_FROM_KEY_STORAGE as a signal KEY_FROM_KEY_PORT. A signal KEY_PORT_BUSY turns on during use of the key access port 62, while it turns off when the storage key is output.

Figure 9 is a block diagram showing the constitution of the control device 61. A signal LA shows the logical address that is transmitted from the CPU instruction computation unit 31. An inverter 91 reverses the main memory access demand signal output from the CPU instruction computation unit 31 to be output. An AND gate 92 outputs the AND operation of the logical address LA and the main memory access demand. An AND gate 93 outputs the AND operation of the output of an OR gate 94 that is stored in a register 95 and the output of the inverter 91. The OR gate 94 outputs the OR operation of the outputs of the AND gates 92 and 93, as signals

LA_TO_TLB, LA_TO_CACHE, LA_TO_KEY_BUFFER, and LA_TO_DAT.

According to such circuit constitution, when the main memory access demand signal turns on, and the logical address is delivered, the delivered logical address is sent to each of TLB 37, the cache 34, the key buffer 38, and DAT 39. Even if the main memory access demand signal turns off after that, the logical address can be transmitted repeatedly since it is stored in the register 95.

Inverters 96 and 97 respectively reverse signals DAT_BUSY and KEY_PORT_BUSY to be output. An AND gate 92 outputs a signal TLB_MISS, and the OR operation of the outputs of the inverters 96 and 97, as an address translation demand.

Further, an address translation completion signal output from DAT 39 is output to the key storage 41 as a key demand signal. The key storage 41 that receives the key demand signal returns a key demand reception signal、and this signal is then output as a TLB registration demand signal. When the registration of the translation result executed by DAT 39 completes, TLB 37 outputs a TLB registration completion signal which turns a signal DAT_BUSY off, thereby releasing DAT 39.

According to such circuit constitution, a key

demand is issued immediately after the completion of address translation. This issuance immediately triggers the TLB registration and the DAT release.

Further, a key arrival report signal output from the key storage 41 is output as a key registration demand signal. The key buffer 38 outputs a key registration completion signal when the registration of a storage key completes. The control device 61 resumes a process of a main memory access demand when it receives the key registration completion signal.

An inverter 99 reverses the key demand signal to be output. An OR gate 100 reverses the OR operation of the denials of the outputs of the inverters 97 and 99 to be output. An AND gate 101 outputs the AND operation of the output of the OR gate 100 and a key registration demand signal, as a signal KEY_PORT_VALID.

As stated previously, two data paths for instruction fetch and for operand fetch exist between the CPU instruction computation unit 31 and the CPU memory unit 32. Two pieces of TLBs 37, key buffers 38, and caches 34 which are provided in the CPU memory unit 32 are installed corresponding to two data paths. Therefore, DAT 39 needs to recognize from which of the instruction fetch

and

the operand fetch an address translation demand is issued.

Thereupon, DAT 39 receives an address translation demand signal for each of instruction fetch and operand fetch, and outputs an address translation completion signal as shown in Figure 10. The control device 61 generates a signal KEY_PORT_VALIDITY for each of the instruction fetch and the operand fetch as shown in Figure 11.

In this case, one of the two key access ports 62 is used for instruction fetch, while the other is used for operand fetch as shown in Figure 12. The key access ports 62 receive a signal KEY_PORT_VALID regarding the instruction fetch or the operand fetch, and outputs a signal KEY_PORT_BUSY.

An AND gate 111 outputs the AND operation of the signal KEY_PORT_BUSY and a key demand signal which are output from the key access port 62 used for instruction fetch. An AND gate 112 outputs the AND operation of the signal KEY_PORT_BUSY and a key demand signal which are output from the key access port 62 used for operand fetch. Then, an OR gate 113 transmits the OR operation of the outputs of AND gates 111 and 112 to the key storage 41 as a key demand signal _TO_KEY_STORAGE.

Here, the following explanation is the case that two key access ports 62 are provided, but generally two or more key access ports 62 can be installed. As the number of key access ports 62 increases, many key accesses can be simultaneously processed.

Next, the characteristics of the cache control of the present invention are added in reference to Figures 13 to 17.

In the implementation of an instruction, it is assumed that there are a plurality of stages as shown in Figure 13. In this case, the action is carried out by the following procedures, in order that CPU implements one instruction.

(1) Instruction fetch unit

T1: Generating an instruction address

T2: Transforming the instruction address (virtual address→physical address)

T3: Reading out an instruction for the instruction address

T4: Checking and decoding an operation code

(2) Instruction execution unit

T5: Generating an operand address

T6: Transforming the operand address

T7: Reading out the operand

T8: Execution

T9: Checking the result

T10: Writing the result in a register or a memory device

When such procedures are controlled with a single flow, the fetch of the following instruction I2 is carried out after the implementation of the preceding instruction I1 terminates, as shown in Figure 14. As a method of speeding up the process of successive instructions, pipeline control as shown in Figure 15 is well known. According to the control, the fetch of instruction I2 can be started immediately after the fetch of instruction I1 terminates.

In this case, the implementation of instruction I1 and the fetch of instruction I2 are carried out simultaneously, so that the respective pieces of hardware are required. That is, for the address translation and the reading out of data that exist in both of the instruction fetch and the instruction execution, the respective pieces of hardware are required. Because of this, TLBs, cache devices and the others are respectively provided in the information processing unit of Figure 2B.

As for DAT, however, the number of pieces of necessary hardware greatly increases, thereby increasing the circuit scale when the table for address

translation is provided with each cache. Thereupon, the instruction fetch and the operand fetch share one DAT in the present preferred embodiments.

When a conventional control method used for DAT and the control method of the present invention used for DAT are compared based on the above-mentioned constitution, the results are as shown in Figure 16. In the case that a TLB miss occurs by the instruction fetch after a TLB miss occurs by the operand fetch, a key access (OP-KEY-WAIT) of the operand fetch is carried out following the translation (OP-DAT) of an operand address by DAT, in the conventional control method. Next, DAT is released after the arrival of a storage key, and the translation of an instruction address by DAT (IF-DAT) as well as a key access of the instruction fetch (IF-KEY-WAIT) are carried out.

In the control method of the present invention, to the contrary, DAT is released immediately after the translation OP-DAT, and the accesses OP-KEY-WAIT and IF-DAT are carried out simultaneously, thereby speeding up the whole process.

When the conventional control method and the control method of the present invention, which are used for TLB registration, are compared, the results are as shown in Figure 17. In the case that a TLB miss (OP-ACC1)

occurs by operand fetch, the access OP-KEY-WAIT is carried out following the translation OP-DAT, and the registration of the address translation result in TLB (OP-KEY-WT) and the registration of a storage key in the key buffer (OP-TLB-WT) are carried out after the arrival of the storage key, in the conventional control method. Next, the process of operand fetch (OP-ACC1) is resumed, a TLB hit occurs, and the next operand fetch (OP-ACC2) is started.

In the control method of the present invention, to the contrary, the registration OP-TLB-WT is carried out immediately after the translation OP-DAT. Therefore, the process OP-ACC1 and the operand fetch OP-ACC2 can be started between the registration OP-KEY-WAIT.

According to the present invention, the access to instructions and operands are made efficiently, in the information processing unit provided with key control protection using a storage key.

Especially, in the data transfer processing to a CPU instruction computation unit, a key access is executed and then data is transmitted, according to the conventional method. In the present invention, on the contrary, data is transmitted during the data access, thereby shortening the time required for the data transfer process. Further regarding a demand which does

not require key control protection, the time required for a data transfer process can be similarly shortened, since data is transmitted and a completion report is issued during a key access.

In a TLB registration process, TLB registration is conventionally executed after a key access is carried out, while in the present invention, TLB registration is executed during a key access. Therefore, TLB can be referred to without waiting for the execution of the preceding demand that is being key-accessed, in the case that a successive demand refers to TLB in the super-scalar processor for performing an out-of-order process. The executing time can be shortened.

In a DAT release process, conventionally, DAT is released after a key access has been executed, while in the present invention, DAT is released while a key access is being executed. Therefore, DAT can be activated for a successive DAT demand without waiting for the process of the preceding demand that is being key-accessed, in the super-scalar processor for performing an out-of-order process. Consequently, the time required for an address translation process can be shortened, and the increase of a circuit scale can be restrained, at the same time.

Furthermore, by preparing a plurality of key access

ports, the successive key access can be executed while the preceding key access is carried out. Accordingly, the address translation processes executed by DAT and a series of key access processes can be continuously executed, thereby shortening the processing time.